UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/727,105	11/29/2000	Andrew A. Chien	ENTRPA.006A	6813

20995 7590 06/04/2004

KNOBBE MARTENS OLSON & BEAR LLP
2040 MAIN STREET
FOURTEENTH FLOOR
IRVINE, CA 92614

EXAMINER	
ABRISHAMKAR, KAVEH	
ART UNIT	**PAPER NUMBER**
2131	

DATE MAILED: 06/04/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/727,105	CHIEN ET AL.	
	Examiner	Art Unit	
	Kaveh Abrishamkar	2131	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *29 November 2000*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-39* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-39* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *4,6*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. This action is in response to the communication filed on November 29, 2000. Claims 1 – 39 were originally received for consideration. No preliminary amendments for the claims were filed. Claims 1 – 39 are currently under consideration.

Information Disclosure Statement

2. Initialed and dated copies of the applicant's IDS forms 1449, Paper No. 4 and 6, are attached to the Office action.

Claim Objections

3. Claim 23 is objected to because of the following informalities: On line 15, "identifies" should be spelled "identified." Appropriate correction is required.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the

applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

4. Claims 4, 6-19, 25-38 are rejected under 35 U.S.C. 102(e) as being anticipated by Hollander et al. (U.S. Patent 6,412,071).

Regarding claim 4, Hollander discloses:

A method of securing an application for execution on a computer, the method comprising:

identifying calls that are made by the application to an external routine (Figure 4 step 160, column 7 line 49 – column 8 line 10);

modifying the binary of an application to invoke an interception module (Figure 4 step 168, column 5 lines 40 – 46, column 7 lines 45-50); and

intercepting at least one of the identified calls at the computer (Figure 4 step 160, column 7 lines 49 – 53).

Regarding claim 6, Hollander discloses:

A method of securing an application for execution on a computer, the method comprising:

identifying calls that cause a detrimental effect to the computer or another application (Figure 4 step 160, column 7 line 49 – column 8 line 10);

modifying a binary of the application to invoke an interception module with respect to the identified calls (Figure 4 step 168, column 5 lines 40 – 46, column 7 lines 45-50); and

intercepting at least one of the identified calls (Figure 4 step 160, column 7 lines 49 – 53).

Regarding claim 7, Hollander discloses:

A method of securing an application for execution on a computer, the method comprising:

intercepting at least one call that is made by the application such that a graphical user interface that is displayed by the application is modified (Figure 4 step 160, column 7 lines 49 – 53);

intercepting at least one call that is made by the application program such that requests for machine or user specific information are virtualized (Figure 4 step 160, column 7 lines 49 – 53); and

intercepting at least one call that is made by the application such that the contents of at least one file that is used by the application is encrypted transparently to the application (Figure 4 step 160, column 7 lines 49 – 53).

Regarding claim 11, Hollander discloses:

intercepting at least one call that is made by the application such that a graphical user interface that is displayed by the application is modified (Figure 4 step 160, column 7 lines 49 – 53); and

intercepting at least one call that is made by the application such that the contents of at least one file that is used by the application is encrypted transparently to the application (Figure 4 step 160, column 7 lines 49 – 53).

Regarding claim 14, Hollander discloses:

A program storage device storing instructions that when executed perform the steps comprising:

intercepting at least one call that is made by the application such that a graphical user interface that is displayed by the application is modified (Figure 4 step 160, column 7 lines 49 – 53); and

intercepting at least one call that is made by the application such that the contents of at least one file that is used by the application is encrypted transparently to the application (Figure 4 step 160, column 7 lines 49 – 53).

Regarding claim 15, Hollander discloses:

A method for allowing application programs to execute in non-native environments, the method comprising:

identifying a service that is not provided by a selected operating system (Figure 4 step 160, column 7 line 49 – column 8 line 10); and

modifying a binary of an application to invoke an interception service instead of requesting the service from the selected operating system (Figure 4 step 168, column 5 lines 40 – 46, column 7 lines 45 – 48).

Regarding claim 16, Hollander discloses:

A program storage device storing instructions that when executed perform the steps comprising:

virtualizing an application interface between a first application and an operating system (Figure 4 step 168, column 5 lines 40 – 46, column 7 lines 45 – 49); and

preventing access by a second application or the operating system to data that is used by the first application (column 5 lines 46 – 62).

Regarding claim 17, Hollander discloses:

A program storage device storing instructions that when executed perform the steps comprising:

virtualizing an application interface between a first application and an operating system (Figure 4 step 168, column 5 lines 40 – 46, column 7 lines 45 – 49); and

preventing the first application from accessing the second application (column 5 lines 46 – 62).

Regarding claim 18, Hollander discloses:

A method of securing an application for execution on a computer, the method comprising:

virtualizing an application interface between a first application and an operating system (Figure 4 step 168, column 5 lines 40 – 46, column 7 lines 45 – 49); and

preventing access by a second application or the operating system to data that is used by the first application (column 5 lines 46 – 62).

Regarding claim 25, Hollander discloses:

A system for securing an application for execution on a computer, the system comprising:

means for intercepting at least one call that is made by the application such that a graphical user interface that is displayed by the application is modified;

means for intercepting at least one call that is made by the application program such that requests for machine or user information are virtualized (Figure 4 step 160, column 7 lines 49 – 53); and

means for intercepting at least one call that is made by the application such that the contents of at least one file that is used by the application is encrypted transparently to the application (Figure 4 step 160, column 7 lines 49 – 53).

Regarding claim 28, Hollander discloses:

A system for securing an application for execution on a computer, the system comprising:

means for intercepting at least one call that is made by the application such that a graphical user interface that is displayed by the application is modified (Figure 4 step 160, column 7 lines 49 – 53); and

means for intercepting at least one call that is made by the application such that the contents of at least one file that is used by the application is encrypted transparently to the application (Figure 4 step 160, column 7 lines 49 – 53).

Regarding claim 31, Hollander discloses:

A system for allowing application programs to execute in non-native environments, the system comprising:

means for identifying a service that is not provided by a selected operating system (Figure 4 step 160, column 7 line 49 – column 8 line 10); and

means for modifying a binary of an application to invoke an interception service instead of requesting the service from the selected operating system (Figure 4 step 168, column 5 lines 40 – 46, column 7 lines 45 – 48).

Regarding claim 32, Hollander discloses:

A system for securing an application for execution on a computer, the system comprising:

means for virtualizing an application interface between a first application and an operating system (Figure 4 step 168, column 5 lines 40 – 46, column 7 lines 45 – 49); and

means for preventing access by a second application or operating system to data that is used by the first application (column 5 lines 46 – 62).

Regarding claim 35, Hollander discloses:

A system for securing an application for execution on a computer, the system comprising:

a preprocessor module for identifying calls that are made by the application to at least one external routine, the preprocessor module modifying the application to invoke an interception module in response to the application invoking the external routine (Figure 4 step 160, step 168, column 5 lines 40 – 46, column 7 line 45 – column 8 line 10).

Regarding claim 38, Hollander discloses:

A method of securing an application for execution on a computer, the method comprising:

rewriting the binary of an application thereby preventing the application from:

accessing a predefined set of data (Figure 4 step 160, column 7 line 49 – column 8 line 10);

invoking a predefined set of instructions (Figure 4 step 168, column 5 lines 40 – 46, column 7 lines 45 – 48); and

accessing one or more files that are in one or more predefined directories Figure 4 step 168, column 5 lines 40 – 46, column 7 lines 45 – 48).

Claim 8 is rejected as applied above in rejecting claim 7. Furthermore, Hollander discloses:

The method of claim 7, wherein the machine information includes operating system information (column 7 lines 49 – 63).

Claim 9 is rejected as applied above in rejecting claim 7. Furthermore, Hollander discloses:

The method of claim 7, additionally comprising intercepting at least one call that is made by the application such that the filename of at least one file that is used by the application is encrypted transparently to the application (column 7 lines 49 – 63).

Claim 10 is rejected as applied above in rejecting claim 7. Furthermore, Hollander discloses:

The method of claim 7, additionally comprising modifying a directory structure of a set of files (column 7 lines 49 – 63).

Claim 12 is rejected as applied above in rejecting claim 11. Furthermore, Hollander discloses:

The method of claim 11, additionally comprising intercepting at least one call that is made by the application such that the filename of at least one file that is used by the application is encrypted transparently to the application (column 7 lines 49 – 63).

Claim 13 is rejected as applied above in rejecting claim 11. Furthermore, Hollander discloses:

The method of claim 11, additionally comprising modifying a directory structure of a set of files (column 7 lines 49 – 63).

Claim 19 is rejected as applied above in rejecting claim 18. Furthermore, Hollander discloses:

The method of claim 18, additionally comprising restricting access by the application to selected resources on the computer (column 5 lines 46 – 62).

Claim 26 is rejected as applied above in rejecting claim 25. Furthermore, Hollander discloses:

The system of claim 25, additionally comprising means for intercepting at least one call that is made by the application such that the filename of at least one file that is used by the application is encrypted transparently to the application (column 7 lines 49 – 63).

Claim 27 is rejected as applied above in rejecting claim 25. Furthermore, Hollander discloses:

The system of claim 25, additionally comprising means for modifying a directory structure of a set of files (column 7 lines 49 – 63).

Claim 29 is rejected as applied above in rejecting claim 28. Furthermore, Hollander discloses:

The system of claim 28, additionally comprising intercepting at least one call that is made by the application such that the filename of at least one file that is used by the application is encrypted transparently to the application (column 7 lines 49 – 63).

Claim 30 is rejected as applied above in rejecting claim 28. Furthermore, Hollander discloses:

The system of claim 28, additionally comprising means for modifying a directory structure of a set of files (column 7 lines 49 – 63).

Claim 33 is rejected as applied above in rejecting claim 32. Furthermore, Hollander discloses:

The system of claim 32, wherein virtualizing the identified calls at the computer comprises virtualizing file system requests (column 5 lines 46 – 62).

Claim 34 is rejected as applied above in rejecting claim 32. Furthermore, Hollander discloses:

The system of claim 32, additionally comprising means for restricting access by the application to selected resources on a computer (column 5 lines 46 – 62).

Claim 36 is rejected as applied above in rejecting claim 35. Furthermore, Hollander discloses:

The system of claim 35, wherein the preprocessor module encrypts at least a portion of a filename that is associated with the application (column 7 lines 49 – 63).

Claim 37 is rejected as applied above in rejecting claim 35. Furthermore, Hollander discloses:

The system of claim 35, wherein the preprocessor module encrypts the contents of at least a portion of the application (column 7 lines 49 – 63).

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. Claims 1-3, 5, 20-24 and 39 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hollander et al. (U.S. Patent 6,412,071) in view of Frese, II et al. (U.S. Patent 5,909,545).

Regarding claim 1, Hollander discloses:

A system for securing an application for execution on a computer, the system comprising:

a preprocessor module for identifying calls that are made by the application to at least one routine that is provided by an operating system, the preprocessor module modifying the application such that an interception module is invoked in response to the application invoking the identified routines (Figure 4 step 160, column 7 line 49 – column 8 line 10).

Hollander does not explicitly describe a server in communication with a client via a network so that the client can receive the modified application from the server before executing the application. Frese discusses a system which includes an application interception module which is transported over a network (Abstract, column 2 lines 1 – 47). The transport of software applications over a network between two computers is well-known in the art and provides the benefit of being able to distribute the security application to a group of computers that have an Internet connection to a server which stores the specific module. Frese address the benefits of reducing the time needed to attain the software (column 2 lines 4 – 6), and the ease of which any computer with an Internet connection can retrieve the application (column 2 lines 13 – 47). Therefore it would have been obvious to one of ordinary skill in the art at the time the applicant's invention was made to use the method of transporting an application module of Frese in conjunction with the module of Hollander to facilitate and expedite the process of loading the application on multiple computers.

Regarding claim 2, Hollander discloses:

A method of securing an application for execution on a computer, the method comprising:

scanning the application program for code sequences that cause the computer to trap the operating system (column 5 lines 22 – 46);

modifying the code sequences such that the computer does not trap to the operating system (column 5 lines 46 – 62);

identifying at least one call that are made by the application to an external routine (Figure 4 step 160, column 7 line 49 – column 8 line 10);

providing at least one interception module for the identified calls (Figure 4 step 168, column 5 lines 40 – 46);

intercepting at least one of the identified calls at the computer (Figure 4 step 160, column 7 lines 49 – 63);

monitoring at the computer usage of resources by the computer (column 3 lines 62 – 67); and

preventing the application from consuming resources in excess of a predefined threshold (column 3 lines 62 – 67, column 6 lines 30 – 35).

Hollander does not explicitly describe a server in communication with a client via a network so that the client can receive the modified application from the server before executing the application. Frese discusses a system which includes an application interception module which is transported over a network (Abstract, column 2 lines 1 – 47). The transport of software applications over a network between two computers is

well-known in the art and provides the benefit of being able to distribute the security application to a group of computers that have an Internet connection to a server which stores the specific module. Frese address the benefits of reducing the time needed to attain the software (column 2 lines 4 – 6), and the ease of which any computer with an Internet connection can retrieve the application (column 2 lines 13 – 47). Therefore it would have been obvious to one of ordinary skill in the art at the time the applicant's invention was made to use the method of transporting an application module of Frese in conjunction with the module of Hollander to facilitate and expedite the process of loading the application on multiple computers.

Regarding claim 3, Hollander discloses:

A method of securing an application for execution on a computer, the method comprising:

scanning the application program for code sequences that cause the computer to trap to the operating system (column 5 lines 22 – 46);

modifying the code sequences such that the computer does not trap to the operating system (column 5 lines 46 – 62);

identifying at least one call that is made by the application to an external routine (Figure 4 step 160, column 7 line 49 – column 8 line 10);

providing at least one interception module for the identified calls (Figure 4 step 168, column 5 lines 40 – 46);

intercepting at least one of the identified calls at the computer (Figure 4 step 160, column 7 lines 49 – 63).

Hollander does not explicitly describe a server in communication with a client via a network so that the client can receive the modified application from the server before executing the application. Frese discusses a system which includes an application interception module which is transported over a network (Abstract, column 2 lines 1 – 47). The transport of software applications over a network between two computers is well-known in the art and provides the benefit of being able to distribute the security application to a group of computers that have an Internet connection to a server which stores the specific module. Frese address the benefits of reducing the time needed to attain the software (column 2 lines 4 – 6), and the ease of which any computer with an Internet connection can retrieve the application (column 2 lines 13 – 47). Therefore it would have been obvious to one of ordinary skill in the art at the time the applicant's invention was made to use the method of transporting an application module of Frese in conjunction with the module of Hollander to facilitate and expedite the process of loading the application on multiple computers.

Regarding claim 20, Hollander discloses:

A system for securing an application for execution on a computer, the system comprising:

means for scanning the application program for code sequences that cause the computer to trap to the operating system (column 5 lines 22 – 46);

means for modifying the code sequences such that the computer does not trap to the operating system (column 5 lines 46 – 62);

means for identifying calls that are made by the application to an external routine (Figure 4 step 160, column 7 line 49 – column 8 line 10);

means for providing at least one interception module for the identified calls (Figure 4 step 168, column 5 line 40 – 46);

means for intercepting at least one of the identified calls at the computer (Figure 4 step 160, column 7 lines 49 – 63);

means for monitoring at the computer the usage of resources by the computer (column 3 lines 62- 67); and

means for preventing the application from consuming resources in excess of a threshold (column 3 lines 62 – 67, column 6 lines 30 – 35).

Hollander does not explicitly describe a server in communication with a client via a network so that the client can receive the modified application from the server before executing the application. Frese discusses a system which includes an application interception module which is transported over a network (Abstract, column 2 lines 1 – 47). The transport of software applications over a network between two computers is well-known in the art and provides the benefit of being able to distribute the security application to a group of computers that have an Internet connection to a server which stores the specific module. Frese address the benefits of reducing the time needed to attain the software (column 2 lines 4 – 6), and the ease of which any computer with an Internet connection can retrieve the application (column 2 lines 13 – 47). Therefore it

would have been obvious to one of ordinary skill in the art at the time the applicant's invention was made to use the method of transporting an application module of Frese in conjunction with the module of Hollander to facilitate and expedite the process of loading the application on multiple computers.

Regarding claim 22, Hollander discloses:

A system for securing an application for execution on a computer, the system comprising:

means for scanning the application program for code sequences that cause the computer to trap to the operating system (column 5 lines 22 – 46);

means for modifying the code sequences such that the computer does not trap to the operating system (column 5 lines 46 – 62);

means for identifying calls that are made by the application to an external routine (Figure 4 step 160, column 7 line 49 – column 8 line 10);

means for providing at least one interception module for the identified calls (Figure 4 step 168, column 5 lines 40 – 46, column 7 lines 45 – 49);

means for intercepting at least one of the identified calls at the computer (Figure 4 step 160, column 7 lines 49 – 63).

Hollander does not explicitly describe a server in communication with a client via a network so that the client can receive the modified application from the server before executing the application. Frese discusses a system which includes an application interception module which is transported over a network (Abstract, column 2 lines 1 –

47). The transport of software applications over a network between two computers is well-known in the art and provides the benefit of being able to distribute the security application to a group of computers that have an Internet connection to a server which stores the specific module. Frese address the benefits of reducing the time needed to attain the software (column 2 lines 4 – 6), and the ease of which any computer with an Internet connection can retrieve the application (column 2 lines 13 – 47). Therefore it would have been obvious to one of ordinary skill in the art at the time the applicant's invention was made to use the method of transporting an application module of Frese in conjunction with the module of Hollander to facilitate and expedite the process of loading the application on multiple computers.

Regarding claim 24, Hollander discloses:

A system for securing an application for execution on a computer, the system comprising:

means for identifying calls that are made by the application to an external routine; means for providing at least one interception module for the identified calls (Figure 4 step 160, column 7 line 49 – column 8 line 10);

means for intercepting at least one of the identified calls at the computer (Figure 4 step 160, column 7 lines 49 – 63).

Hollander does not explicitly describe a server in communication with a client via a network so that the client can receive the modified application from the server before executing the application. Frese discusses a system which includes an application

interception module which is transported over a network (Abstract, column 2 lines 1 – 47). The transport of software applications over a network between two computers is well-known in the art and provides the benefit of being able to distribute the security application to a group of computers that have an Internet connection to a server which stores the specific module. Frese address the benefits of reducing the time needed to attain the software (column 2 lines 4 – 6), and the ease of which any computer with an Internet connection can retrieve the application (column 2 lines 13 – 47). Therefore it would have been obvious to one of ordinary skill in the art at the time the applicant's invention was made to use the method of transporting an application module of Frese in conjunction with the module of Hollander to facilitate and expedite the process of loading the application on multiple computers.

Claim 5 is rejected as applied above in rejecting claim 4. Hollander does not explicitly describe a server in communication with a client via a network so that the client can receive the modified application from the server before executing the application. Frese discusses a system which includes an application interception module which is transported over a network (Abstract, column 2 lines 1 – 47). The transport of software applications over a network between two computers is well-known in the art and provides the benefit of being able to distribute the security application to a group of computers that have an Internet connection to a server which stores the specific module. Frese address the benefits of reducing the time needed to attain the software (column 2 lines 4 – 6), and the ease of which any computer with an Internet connection

can retrieve the application (column 2 lines 13 – 47). Therefore it would have been obvious to one of ordinary skill in the art at the time the applicant's invention was made to use the method of transporting an application module of Frese in conjunction with the module of Hollander to facilitate and expedite the process of loading the application on multiple computers.

Claim 21 is rejected as applied above in rejecting claim 20. Furthermore, Hollander discloses:

The system of claim 20, wherein the threshold is determined in real time by monitoring the system state (column 3 lines 62 – 67).

Claim 23 is rejected as applied above in rejecting claim 22. Furthermore, Hollander discloses:

The system of claim 22, wherein the means for intercepting at least one of the identified calls prevents the application from communicating with network devices that are not listed in a pre-approved list of network connections (column 7 lines 49 – 63).

Claim 39 is rejected as applied above in rejecting claim 38. Furthermore, Hollander discloses:

The method of claim 38, additionally comprising rewriting the binary of the application thereby preventing the application from modifying an output device of the computer (Figure 4 step 168, column 5 lines 40 – 46, column 7 lines 45 – 49).

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Kaveh Abrishamkar whose telephone number is 703-305-8892. The examiner can normally be reached on Monday thru Friday 8-5.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Ayaz Sheikh can be reached on 703-305-9648. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is 703-305-3900.

KA
5/28/04

AYAZ SHEIKH
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100